## Remarks

Claims 1, 3-8, and 10-21 remain in this application. Claims 2 and 9 were previously canceled without prejudice. Claims 1, 8, and 15 are currently amended. Claim 16-21 are newly added. No new matter has been added.

## Claim Rejections--35 USC 102

Original claims 1, 4, 5, 7, 8, 11, 12, 14 and 15 were rejected under 35 U.S.C. § 102 as being anticipated by Ayers. These rejections are respectfully traversed with respect to the claims as now amended.

Claim 1, as currently amended, recites as follows.

1. A method of compiling a computer program, the method comprising:
   receiving a plurality of modules of source code;
   generating intermediate representations corresponding to the modules;
   extracting a set of data from the intermediate representations to create an inliner summary for each module;
   using the inliner summaries and a globally-sorted working-list based order in an inline analysis phase, **without using the intermediate representations in the inline analysis phase,** to determine which call sites in the modules are to be inlined by substituting code from a called module, wherein **said globally-sorted working-list based order is dynamically updated during the inline analysis phase;** and
   after a call site is determined to be inlined, updating a call graph of the routines and call sites, and updating the inliner summaries throughout the call graph.

(Emphasis added.)

First, applicants respectfully submit that the claim limitation of "using the inliner summaries and a globally-sorted working-list based order in an inline analysis phase, **without using the intermediate representations in the inline analysis phase,** to determine which call sites in the modules are to be inlined by substituting code from a called module" is contrary to the disclosure and teachings of Ayers.

10/699,144
Amendment and Response to Office Action

Atty. Docket No. 200313003-1
October 3, 2007

This claim limitation is supported, for example, in the original application at page 3. lines 11-29, which is reproduced below for convenience of reference.

> **In accordance with an embodiment of the invention, the intermediate representation is never used by the inline analyzer. The inline analyzer uses summary information alone.** The effect of an inline is captured by updating summaries after each and every inline. The propagation of summaries is done elaborately, potentially throughout the call-graph. Existing state-of-the-art frameworks lack proper updation when summaries are used. A formal method is devised to compute the goodness of call sites with a view to comparing them and ordering them in a descending order of goodness. The novelty of computing the goodness lies in its derivation from the summary information alone, unlike existing schemes. **The compile-time effectiveness of our invention lies in the ability to work consistently with summary information in the inline analysis phase without having to touch the intermediate representation,** while maintaining a high degree of run-time performance by continuously updating the summary information. One particular embodiment of the invention lies in the use of a global worklist approach wherein all the call sites are examined in order depending on their goodness factors. Applicants believe this is a novel approach not employed hitherto by any other scheme. Unlike often used bottom-up and top-down approaches, this novel framework has the potential to obtain the best run-time performance.

(Emphasis added.)

In other words, the summary information is a "light weight" representation which extracts a subset of information from the comprehensive or "heavy weight" intermediate representation.

In contrast, Ayers teaches using the full or "heavy weight" intermediate representation to perform the inline analysis. This is supported by the following citation. Ayers, section 2.2, first paragraph, states as follows. "Conceptually, HLO operates as something of a pipeline. <u>The input stage translates the ucode into HLO's own internal representation</u>, and builds up a comprehensive symbol table. <u>A variety of classic optimizations (e.g. constant propagation) are performed on the IR at this time, mainly to reduce its size. After all code has been input, a limited amount of interprocedural analysis is performed. HLO then inlines</u> and clones in a manner we describe below. The output phase converts the HLO IR back into ucode and send the ucode on to the back end for intensive interprocedural optimization and ultimately generation of object code." (Emphasis added.)

Therefore, Ayers teaches use of the IR in performing inline analysis which is contrary to this claim limitation.

Second, applicants respectfully submit that the claim limitation that "**said globally-sorted working-list based order is dynamically updated during the inline analysis phase**" is contrary to the disclosure and teachings of Ayers. This claim limitation is supported, for example, in the original application at page 23, lines 29-32, which is reproduced below for convenience of reference.

> If the caller or the callee routine for a certain call site has its summary information modified, the goodness factor of that call site is recomputed. **Whenever the goodness factor of a certain call site is recomputed, the old entry is deleted from the working list and a new updated entry is inserted.**

(Emphasis added.)

In other words, the working list for the inline analysis phase is dynamically updated based on changes in the goodness factor of call sites.

In contrast, Ayers teaches using an initial (statically sorted) list throughout to perform the inline analysis. This is supported by the following citation. Ayers, section 2.4, third paragraph, lines 1-4, states as follows. "<u>The inliner then walks over the inline site list in priority order.</u> The compile-time impact of each site is considered, and if within the current budget, the inline is accepted." (Emphasis added.)

Therefore, Ayers teaches going through the inline site list in a static "priority" order. This is contrary to this claim limitation.

Thus, for at least the above discussed reasons, applicants respectfully submit that amended claim 1 now overcomes this rejection.

Claims 4, 5 and 7 depend from amended claim 1. Therefore, applicants respectfully submit that claims 4, 5 and 7 now also overcome this rejection for at least the same reasons as discussed above in relation to amended claim 1.

Further regarding claim 4, claim 4 recites "wherein updating the inliner summaries comprises determining nodes and edges of the call graph that are affected by the inlining of the call site and updating those inliner summaries corresponding to the affected nodes and edges." In contrast, Ayers states at section 2.3, the last paragraph, lines 5-8 as follows. "The call site is then modified to refer to the clone instead of the original routine. This modification in turn inspires changes in

the call graph to reflect the new relationships between caller, clonee, and clone." There are at least two distinctions between claim 4 and this citation to Ayers. First, claim 4 pertains to updating summaries of the affected nodes and edges, while Ayers does not even maintain such summary information. Second, the citation to Ayers is describing the update for cloning, not inlining.

Further regarding claim 5, claim 5 recites "wherein the edge summaries include at least a call site execution count and a signature type." In contrast, Ayers states at section 2.3, first paragraph, "Cloning begins with the selection of cloning sites. Each call site is examined in turn. The cloner first determines if the call site passes certain legality tests. For example, cloning is disallowed if there are gross type mismatches between caller and callee, or if the caller and callee do not agree on the number of parameters to be passed." There are at least three distinctions between claim 5 and this citation to Ayers. First, while claim 5 recites components of the inliner summaries, Ayers does not even maintain summary information. Second, Ayers does not disclose or teach "a call site execution count" or a "signature type". Third, Ayers is describing this process for cloning, not inlining.

Regarding claim 7, claim 7 requires that the inline analysis phase is separate and distinct from an inline transformation phase. We respectfully submit that this claim is not anticipated by Ayers. Please note that Sec. 2.3 in Ayers refers to cloning while claim 7 specifically recites "inline" analysis and transformation phases.

Similarly as with claim 1, claim 8 is amended so as to now recite "a **dynamically-updated** globally-sorted working-list based order to analyze the call sites in an inline analysis phase, **without using the intermediate representations**, so as to determine which call sites in the modules are to be inlined ...." Therefore, applicants respectfully submit that amended claim 8 now also overcomes this rejection for at least the same reasons as discussed above in relation to claim 1.

Claims 11, 12 and 14 depend from amended claim 8. Therefore, applicants respectfully submit that claims 11, 12 and 14 now also overcome this rejection for at least the same reasons as discussed above in relation to amended claim 8.

10/699,144
Amendment and Response to Office Action

Atty. Docket No. 200313003-1
October 3, 2007

Furthermore, claim 11 further recites "wherein the cross-module optimizer is further configured to update the inliner summaries by determining nodes and edges of the call graph that are affected by the inlining of the call site and update those inliner summaries corresponding to the affected nodes and edges." In contrast, Ayers states at sec. 2.3, the last paragraph, lines 5-8 as follows. "The call site is then modified to refer to the clone instead of the original routine. This modification in turn inspires changes in the call graph to reflect the new relationships between caller, clonee, and clone." There are some important distinctions between claim 11 and this citation to Ayers. First, claim 11 pertains to updating summaries of the affected nodes and edges, while Ayers does not even maintain such summary information. Second, the citation to Ayers is describing the update for cloning, not inlining.

Further regarding claim 12, claim 12 recites "wherein the edge summaries generated by the front-end portion include at least a call site execution count and a signature type." In contrast, Ayers states at section 2.3, first paragraph, "Cloning begins with the selection of cloning sites. Each call site is examined in turn. The cloner first determines if the call site passes certain legality tests. For example, cloning is disallowed if there are gross type mismatches between caller and callee, or if the caller and callee do not agree on the number of parameters to be passed." There are at least three distinctions between claim 12 and this citation to Ayers. First, while claim 12 recites components of the inliner summaries, Ayers does not even maintain summary information. Second, Ayers does not disclose or teach "a call site execution count" or a "signature type". Third, Ayers is describing this process for cloning, not inlining.

Regarding claim 14, claim 14 requires that the inline analysis phase is separate and distinct from an inline transformation phase. We respectfully submit that this claim is not anticipated by Ayers. Please note that Sec. 2.3 in Ayers refers to cloning while claim 14 specifically recites "inline" analysis and transformation phases.

Also similarly as with claim 1, claim 15 is amended and now recites "a **dynamically-updated** globally-sorted working-list based order to analyze the call sites in an inline analysis phase, **without using the intermediate representations,**

to determine which call sites in the modules are to be inlined ...." Therefore, applicants respectfully submit that amended claim 15 now also overcomes this rejection for at least the same reasons as discussed above in relation to claim 1.

New claim 16 recites "using inliner summaries in a one-pass inline analysis phase, **without using the intermediate representations**, to determine which call sites to inline" and "**dynamically updating** the work-list every time a call site is accepted for inlining." (Emphasis added.) Therefore, applicants respectfully submit that amended claim 16 now also overcomes this rejection for at least the same reasons as discussed above in relation to claim 1. New claims 17-21 depend from claim 16.

## Claim Rejections--35 USC 103

Original claims 3, 6, 10 and 13 were rejected under 35 U.S.C. § 103 as being unpatentable over Ayers in view of Schmidt. These rejections are respectfully traversed with respect to the claims as now amended.

Claims 3 and 6 depend from amended claim 1. Therefore, applicants respectfully submit that claims 3 and 6 now also overcome this rejection for at least the same reasons as discussed above in relation to Ayers and amended claim 1.

Claims 10 and 13 depend from amended claim 8. Therefore, applicants respectfully submit that claims 10 and 13 now also overcome this rejection for at least the same reasons as discussed above in relation to Ayers and amended claim 8.

Furthermore, regarding claim 3, claim 3 recites "after the call graph and inliner summaries are updated, re-calculating profitabilities associated with remaining call sites; and re-ordering the working list using the re-calculated profitabilities." We respectfully submit that this claim is not anticipated from the teachings of Ayers and Schmidt.

The following is quoted from Schmidt, col. 7 lines 31-38: "If the call site exceeds the allowable growth at decision block 408, its priority is recalculated based on the bloat now known that will be incurred as indicated at a block 410. A best call

site j residing in the AuxQueue is obtained as indicated at a block 412. Then checking to see whether the best call site j residing in the AuxQueue obtained at block 412 has a priority at least as great as the one being considering is performed as indicated at a decision block 414." The following is quoted from Schmidt, col. 7 lines 39-51: "If the priority of the best call site j is less than or equal to the calculated priority at block 412, then checking is performed to determine whether the alternate best call site j from the AuxQueue should be inlined instead as indicated at a decision block 416. If the bloat from the alternate call site would also exceed what would have been acceptable for the original call site, the alternate call site from the AuxQueue should not be inlined either at this time. Otherwise, when determined that the bloat estimate for the alternate call site j from the AuxQueue is less than the bloat estimate i multiplied by the growth threshold at decision block 416, then the original candidate i is set to false as indicated at a block 418."

There are at least two distinctions between claim 3 and Schmidt. First, in contrast to Schmidt, the claimed method operates on summary information. Second, claim 3 requires re-ordering the working list. Schmidt does not do such re-ordering; it has a number of rounds of inlining and no re-computation of profitabilities is done in the initial round. As seen above in Schmidt, col. 7, lines 39-51, in the second round, the technique traverses from the leaves towards the root of the call-graph and compares the profit of the already-accepted call-sites with the profit from the elements in AuxQueue. The elements in AuxQueue are the not-already-inlined call sites that were encountered during this upward traversal. Depending on the comparison, Schmidt refines the decisions made in the initial round. Thus, "Schmidt" is not performing any real re-sorting, only refinement. (This reiterates our claim per new claim 16 that our technique is the only one that performs the analysis process in a single round and that it does so by continuously and dynamically re-sorting the work-list.)

Further regarding claim 10, claim 10 recites similar limitations as claim 3, and so claim 10 is distinguished over Ayers and Schmidt for at least the above-discussed reasons in relation to claim 3.

10/699,144
Amendment and Response to Office Action

Atty. Docket No. 200313003-1
October 3, 2007

## New Claims

Support for the elements of new claims 16-21 are found in the specification at locations indicated in parenthesis and italics below.

16. A method of compiling a computer program with a plurality of modules of source code and corresponding intermediate representations, the method comprising:

    extracting a set of data from the intermediate representations of the modules to create a light-weight inliner summary for each module; *(page 3, lines 11-13)*

    using the inliner summaries in a one-pass inline analysis phase, without using the intermediate representations, to determine which call sites to inline, in what order to inline them, and preserving a same order during the transformation phase; *(page 10, lines 15-20; page 29, lines 10-13)*

    formulating a measure of goodness for each call site from the light-weight inliner summary; *(page 3, lines 16-20; page 8, lines 19-21; page 25, lines 1-14)*

    using a technique to dynamically update information in the light-weight summary for potentially all call-graph nodes and edges every time a call site is accepted for inlining; and *(page 5, lines 31-33; page 6, lines 1-4; page 21, starting from line 19)*

    using a globally sorted work-list and an associated table to maintain and manipulate the call sites and dynamically updating the work-list every time a call site is accepted for inlining. *(page 23, lines 29-32)*

17. The method of claim 16, wherein the inliner summaries are comprised of: a code size; a call site profile count; a critical path length; an execution time; a node level; a call savings; an optimization savings; a level criticality; a normalized level; and a total execution count. *(Section entitled "Summary Information" starting on page 11.)*

10/699,144
Amendment and Response to Office Action

Atty. Docket No. 200313003-1
October 3, 2007

18. The method of claim 16, wherein an arbitrary inlining order among the call sites in the call graph is selectable in an inline analysis phase, and wherein that same order is followed in an inline transformation phase. *(Page 8, line 7 through page 10, line 10.)*

19. The method of claim 16, wherein a measure of goodness for each call site is computed from the light-weight inliner summary, and a total profit is computed as a product of component profit factors, and a total cost is computed as a product of component cost factors. *(Equation at the top of page 25.)*

20. The method of claim 16, wherein information in the light-weight inliner summary corresponding to call-graph nodes and edges are updated each time a call site is accepted for inlining, and wherein a goodness factor of each call site with updated summary information is re-computed. *(Section entitled "Summary Updation" starting on page 21.)*

21. The method of claim 16, wherein a globally-sorted work list and an associated table are maintained and used to continuously order the work list and extract a call site with a highest goodness factor. *(Page 23, lines 29-32)*

10/699,144
Amendment and Response to Office Action

Atty. Docket No. 200313003-1
October 3, 2007

## Conclusion

For the above-discussed reasons, applicant respectfully submits that the pending claims, as hereby amended, now overcome the rejections of the latest office action and are now in form for allowance. Favorable action is respectfully requested.
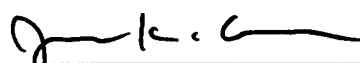
The Examiner is also invited to call the below-referenced attorney to discuss this case.
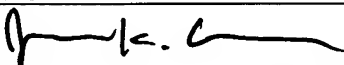
Respectfully Submitted,

Dhruva R. Chakrabarti et al.

Dated: 10/3/2007

James K. Okamoto, Reg. No. 40,110
Tel: (408) 436-2111
Fax: (408) 436-2114

| CERTIFICATE OF MAILING | | | |
|---|---|---|---|
| I hereby certify that this correspondence, including the enclosures identified herein, is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on the date shown below. If the Express Mail Mailing Number is filled in below, then this correspondence is being deposited with the United States Postal Service "Express Mail Post Office to Addressee" service pursuant to 37 CFR 1.10. | | | |
| Signature: | | | |
| Typed or Printed Name: | James K. Okamoto | Dated: | 10/3/2007 |
| Express Mail Mailing Number (optional): | | | |